# MAS 622J: Pattern Recognition and Analysis

**Problem Set 6**

**Date:** Lecture 16,17 **Due:** Lecture 20

1. Implement the EM algorithm to model the given data sets using a mixture of Gaussians. Each of the data sets are 2-dimensional and you can visualize the data using the scatterplot. (For example, *scatter(data1train(:,1), data1train(:,2))*). Experiment with the number of mixtures and try to find a fixed number which works well for all data sets. Comment on the tradeoff between number of mixtures and goodness of the fit to the data. Show your results by plotting each data set superimposed with one-sigma equal density contours of each mixing component. List mean, covariance, and mixing weights of each mixture component. Include a complete listing of your source code.

2. In this problem, we are going to build a classifier to recognize handwritten digits from the OCR data. The 32 x 32 bitmaps of handwritten digits are preprocessed and are divided into nonoverlapping blocks of 4x4 and the number of on pixels are counted in each block. This generates an input matrix of 8x8 where each element is an integer in the range 0..16. This reduces dimensionality and gives invariance to small distortions. The OCR datasets are from the UCI Machine Learning Repository:

   (http://www.ics.uci.edu/ mlearn/MLSummary.html)

   Please see there under *optical character recognition* for references and data source. The dataset was further processed, to rescale the input elements to in between 0-1, and to transform the output values into 10 binary outputs.

   The dataset is available on the course website. Simply do a *load* in matlab on one of the files. You get two arrays, *in* and *out*. *in* contains 8x8 images of characters, making it of size 64 x examples. *out* contains the classifications for the examples in *in*, as binary flags with zeros everywhere except for a 1 in the correct position (i.e. 0 0 0 0 0 0 1 0 0 0 is a 6.)

   For this problem you are free to write your own code or use any MATLAB®toolboxes available for the purpose. You can also use the default Neural Network toolbox available in MATLAB®.

   The following commands might help to get you started with the MATLAB®Neural Net toolbox:*help nnet, help nntool*. If you type *demo* on the MATLAB®prompt, under the option *Toolboxes*, you can select *Neural Networks* to view some examples.

   Train a two-layer neural network with sigmoidal hidden units to perform OCR. Train the network using the back-propagation algorithm with the provided training set. Test your network and report recognition results. Experiment with different numbers of hidden units to optimize recognition accuracy. Comment on the effects of varying the number of hidden units on recognition accuracy. Include a complete listing of your source code.

MATLAB® is a trademark of The MathWorks, Inc.